Decentralized learning over Wireless Networks with broadcasted-based Subgraph Sampling

Supervised by: Jaiprakash Nagar

Team members: Arsalene Khachmadi
Imene Jedidi
Sara Rosato

Table of contents

01

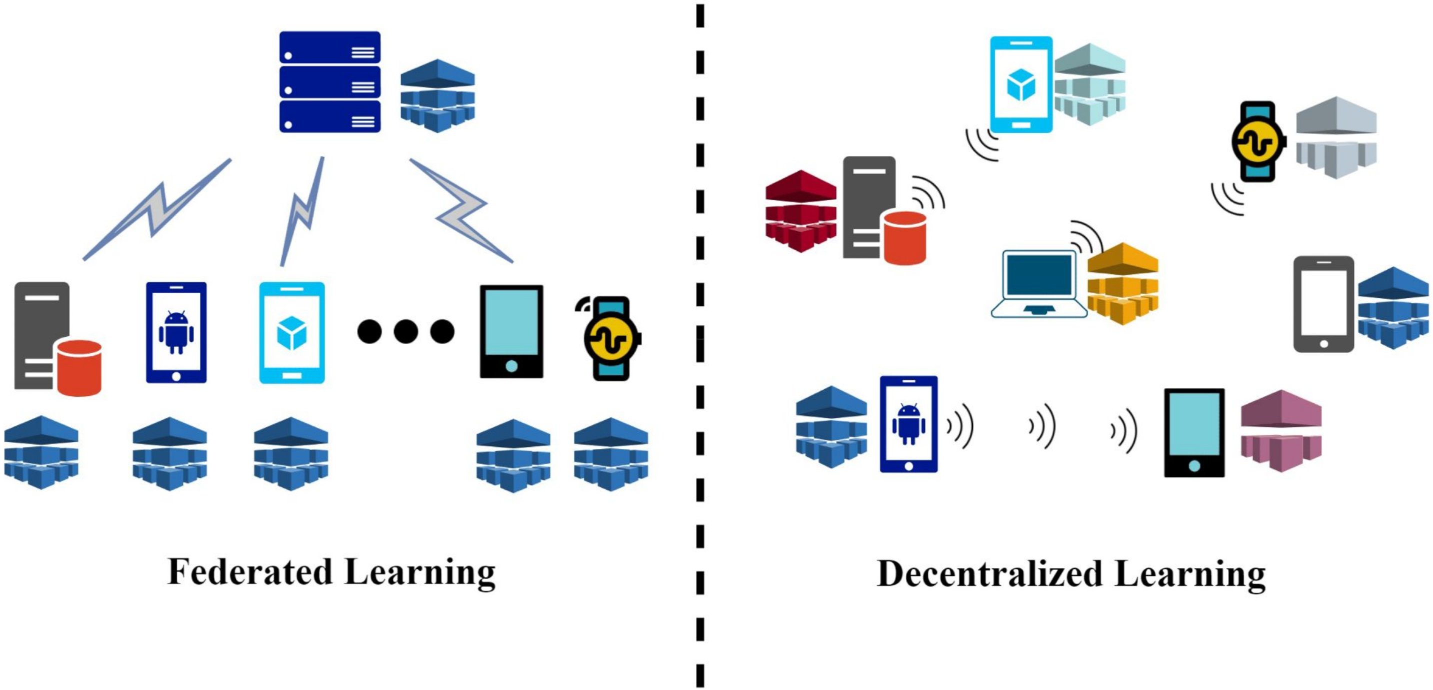
Introduction

Overview of decentralized learning

Decentralized learning involves multiple agents collaboratively training a machine learning model using their local data without sharing the data itself.

Importance:

- Ensures data privacy
- Reduces the need for centralized data storage
- Enhances scalability and robustness in distributed systems



Federated Learning
Decentralized Learning

Objectives of the project

- **Main goal:**

  Duplication of the BASS framework to improve the efficiency of decentralized stochastic gradient descent (D-SGD) in wireless networks.

- **Key challenges addressed:**

  - High communication costs and delays in wireless networks.

  - Issues such as packet collision and access control.

02

BASS Framework Overview

Key Innovations

- Broadcast Transmission: nodes can share information with many other nodes at the same time
- Subgraph Sampling: select smaller groups of nodes to send updates at different times
- Symmetric Communication: makes sure that the connections between nodes work both ways

Advantages of BASS

- Improved convergence
- Scalability
- Resilience to Single-Node Failures

03

State of the Art

Full Communication in D-SGD

Every node communicates with all other nodes in every iteration.

Drawbacks:

- High communication overhead
- Increased communication delays
- Higher energy consumption

MATCHA

- Balances error convergence and runtime
- Decomposes communication topology into matchings

Comparison with BASS:

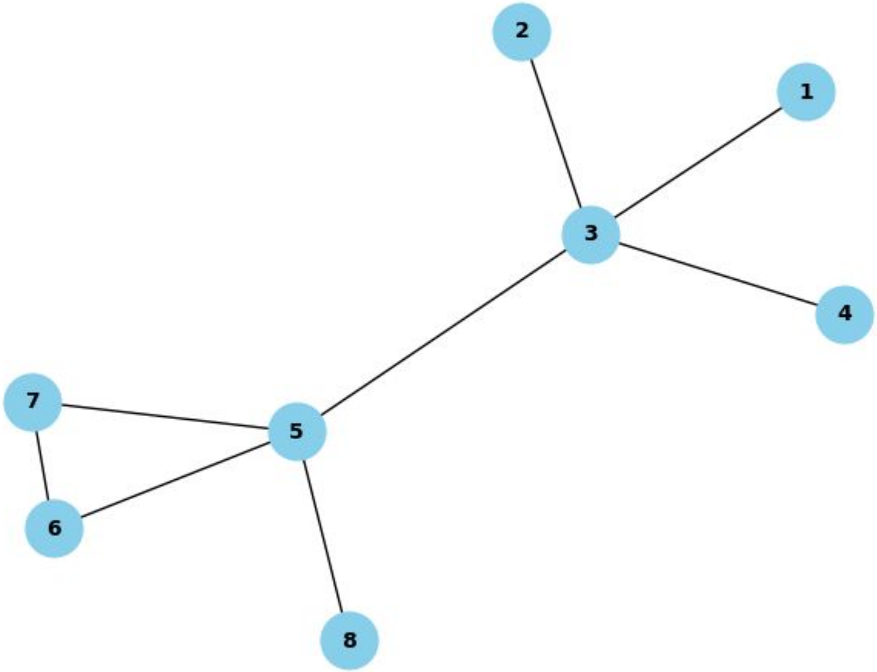
- MATCHA reduces runtime with flexible communication budgets
- BASS improves efficiency and reduces delays through broadcast-based communication

Error-runtime Trade-off:
MATCHA achieves faster training times but has higher communication costs compared to BASS

04

Implementation Details

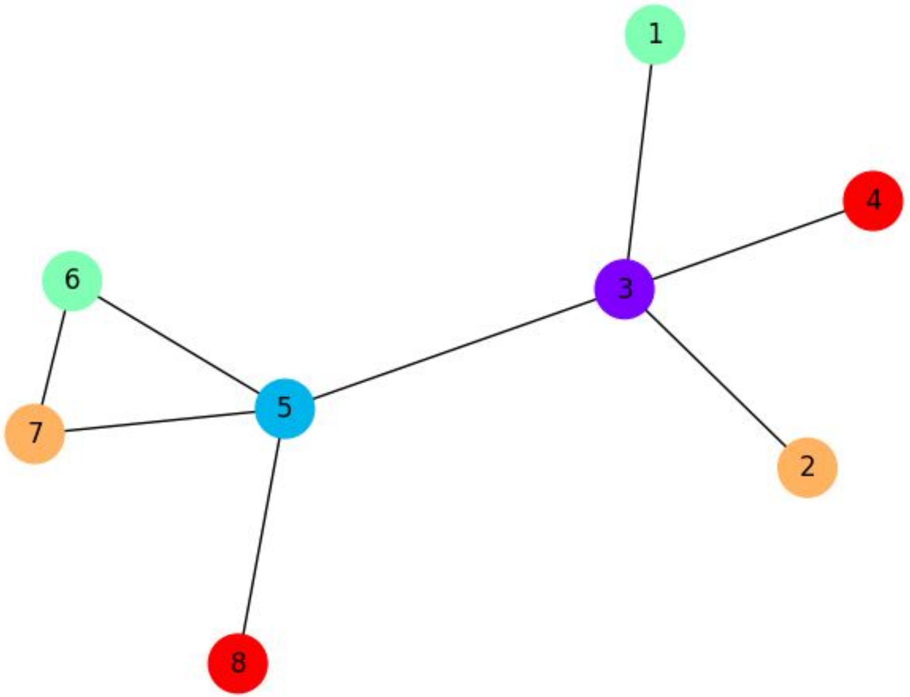
Network Topology



Steps

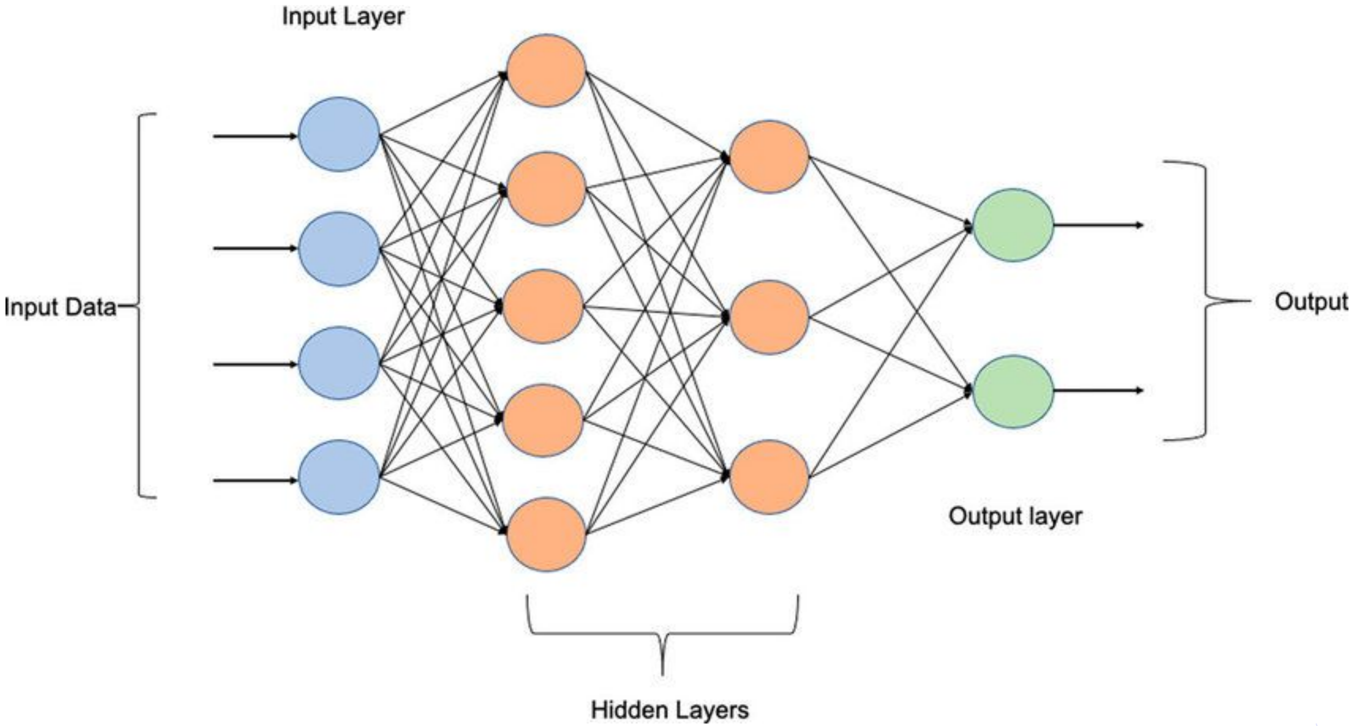
1. Implement base topology
2. Compute betweenness centrality (Node importance)
3. Normalize and assign sampling probabilities
4. Modify Network topology: create auxiliary graph by adding links
5. Apply greedy coloring algorithm: assign colors to nodes in order to create collision-free subsets

Subset division



Subset 0: [3]
Subset 1: [5]
Subset 2: [1, 6]
Subset 3: [2, 7]
Subset 4: [4, 8]

Multilayer Perceptron



05

Experiments

Training Setup

- Dataset: MNIST
- Division: Training and testing datasets
- Training Parameters:
  - Number of epochs: 40
  - Loss function: Sparse Categorical Cross Entropy
  - Optimizer: Adam with decaying learning rate
  - Weight decays tested: [0.0001, 0.0005, 0.001, 0.005]

Results:

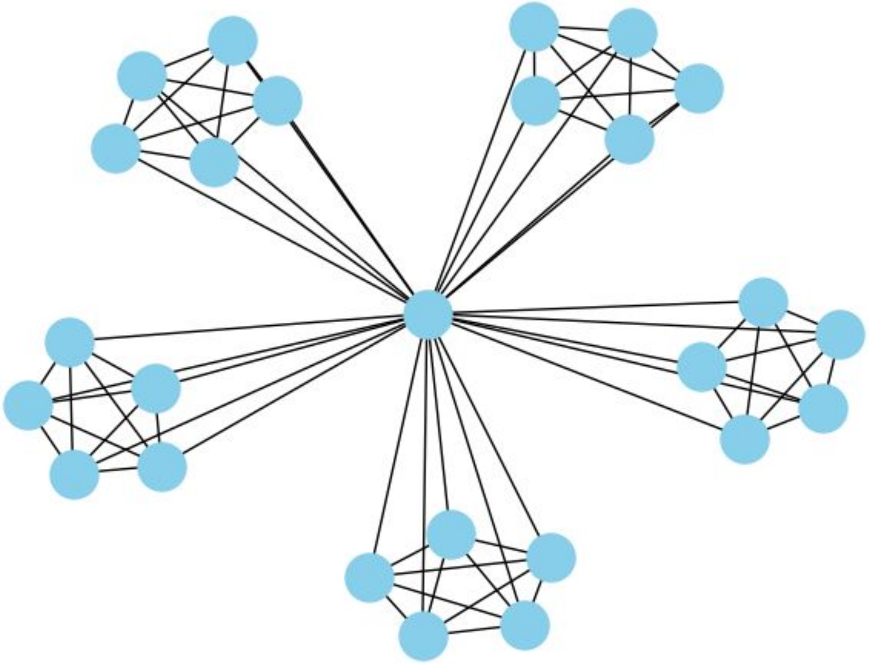
Base topology results:

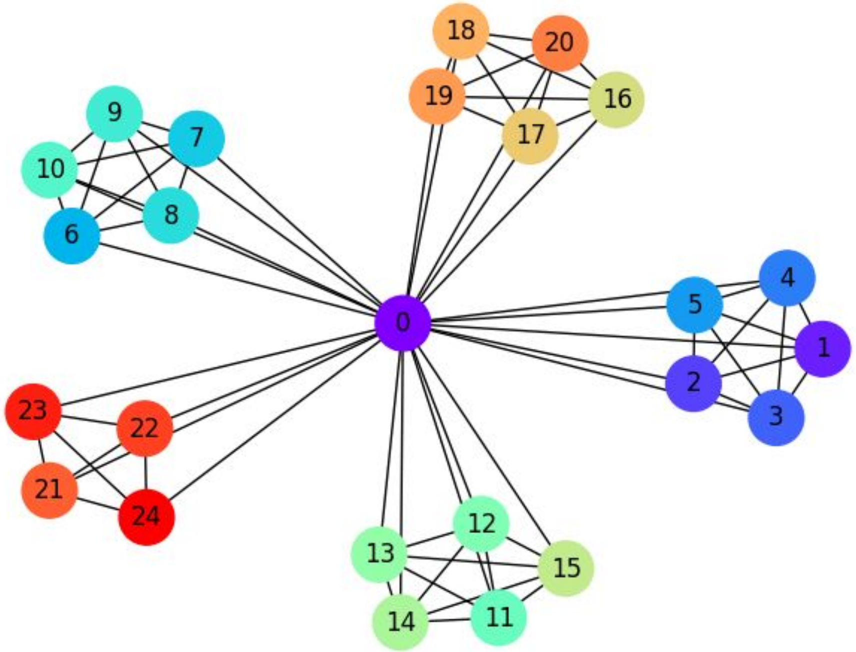
- Test Loss: 0.0741
- Test Accuracy: 99.19%

Topologies compared:

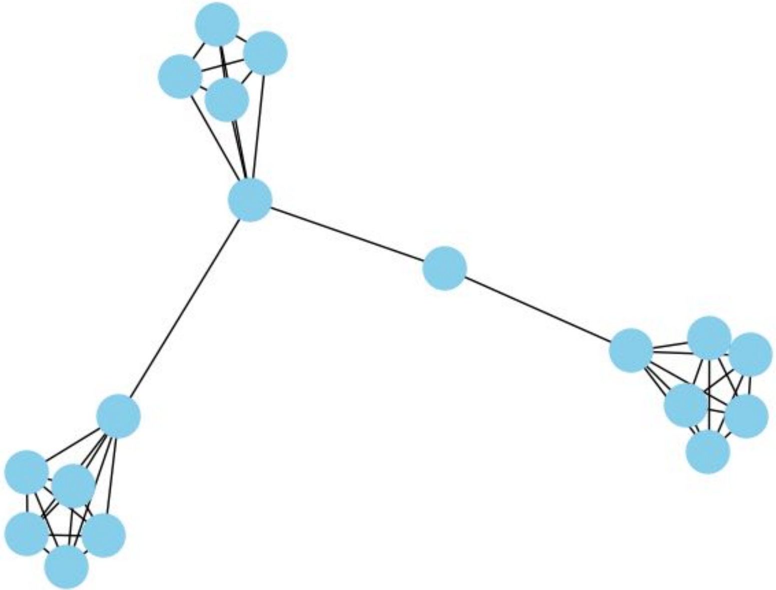
- Star-like (25 nodes)
- Linear (20 nodes)
- Tree-like (15 nodes)

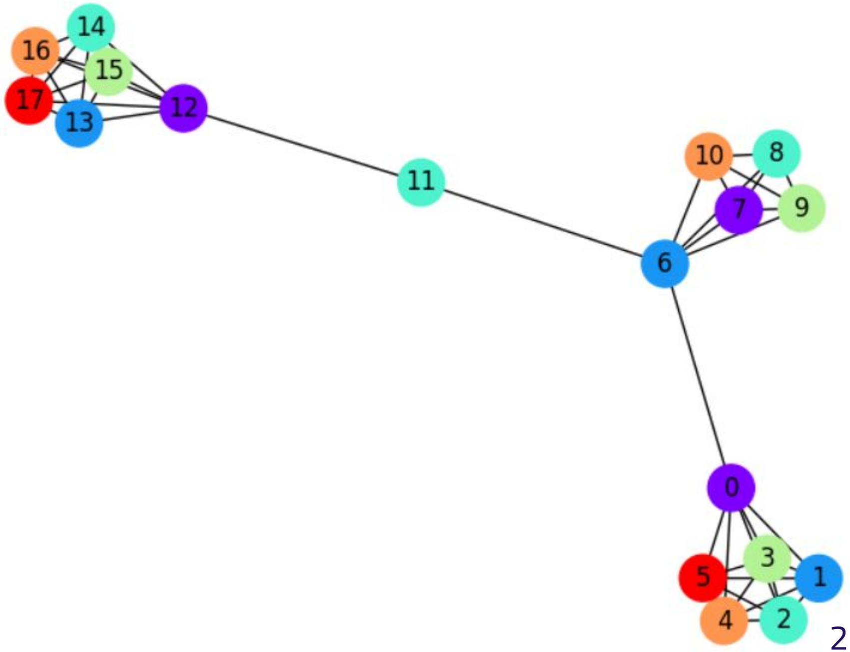
Star topology with N = 25



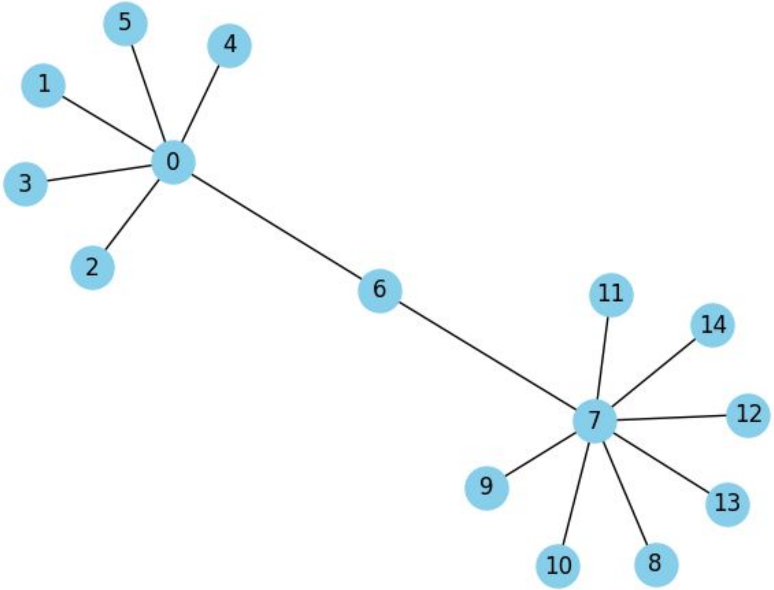


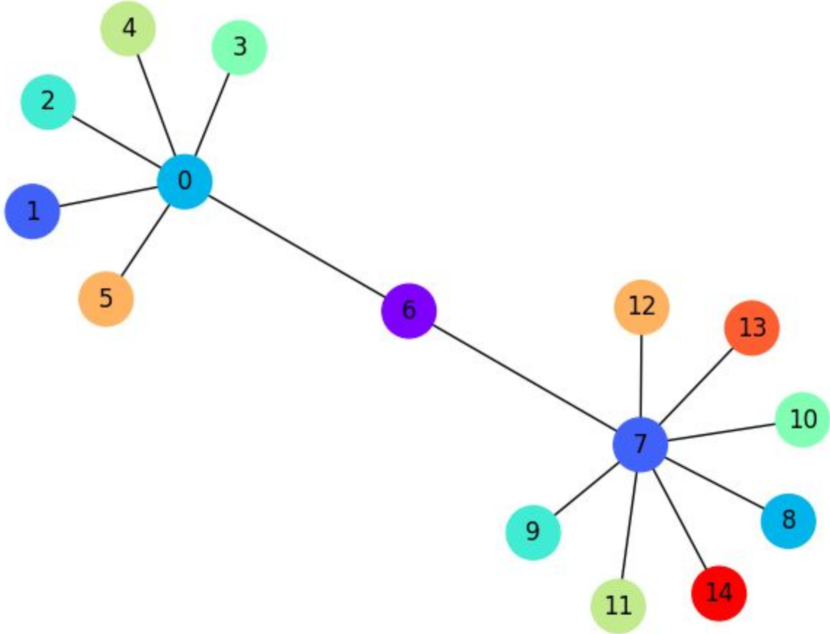
Tree topology with N = 20





Tree topology with N = 15





Results

Graph	Test loss	Accuracy
1	0.1826	96.89%
2	0.1661	97.26%
3	0.1578	97.24%

06

Alternative approaches and conclusions

Alternative approaches

Other Node Importance Metrics:

- Eigenvalue Analysis
- Entropy measurement

Alternative Partitioning methods:

- Clustering algorithms (K-means, Spectral clustering)

Conclusion and perspectives

Summary findings:

- Validation of BASS framework
- Efficiency improvements in decentralized learning

Future work:

- Exploration of alternative methods
- Real-world applications

Thank you for your attention!